

# Evolution of Supershapes for the Generation of Three-Dimensional Designs

Richard J. Preen and Larry Bull  
 richard.preen@live.uwe.ac.uk  
 larry.bull@uwe.ac.uk

University of the West of England, Bristol BS16 1QY, United Kingdom

**Abstract.** This paper explores the evolution of three-dimensional objects with a simple generative encoding, known as the Superformula. Evolving three-dimensional objects has long been of interest in a wide array of disciplines, from engineering (e.g., robotics) to biology (e.g., studying morphological evolution). While many representations have been presented, ranging from direct encodings to complex graphs and grammars, the vast majority have possessed complex underlying encodings, which were necessary to produce varied morphologies. Here we explore the target-based evolution of Superformula, which is simply encoded as a vector of reals and show that it is possible to generate very closely matching designs of a number of complex three-dimensional objects.

## 1 Introduction

The evolution of geometric models to design arbitrary three-dimensional morphologies has been widely explored. Early examples include Watabe and Okino's lattice deformation approach [28] and McGuire's sequences of polygonal operators [17]. Sims [25] evolved the morphology and behaviour of virtual creatures that competed in simulated three-dimensional worlds with a directed graph encoding. Bentley [3] investigated the creation of three-dimensional solid objects via the evolution of both fixed and variable length direct encodings. The objects evolved included tables, heatsinks, penta-prisms, boat hulls, aerodynamic cars, as well as hospital department layouts. Eggenberger [6] evolved three-dimensional multicellular organisms with differential gene expression. Jacob and Nazir [15] evolved polyhedral objects with a set of functions to manipulate the designs by adding stellating effects, shrinking, truncating, and indenting polygonal shapes. More recently, Jacob and Hushlak [14] used an interactive evolutionary approach with L-systems [21] to create virtual sculptures and furniture designs.

Few evolved designs however have been manufactured into physical objects. Conventionally evolved designs tend to be purely descriptive, specifying *what* to build but not *how* it should be built. Thus there is always an inherent risk of evolving interesting yet unbuildable objects. Moreover, high fidelity simulations are required to ensure that little difference is observed once the virtual design is physically manifested. In highly complex design domains such as dynamic

objects, the difference between simulation and reality is too large to manufacture designs evolved under a simulator.

Funes and Pollack [8] performed one of the earliest attempts to physically instantiate evolved three-dimensional designs by placing physical LEGO bricks according to the schematics of the evolved individuals. A direct encoding of the physical locations of the bricks was used and the fitness was scored using a simulator which predicted the stability of the composed structures. Pollack *et al.* [19] subsequently evolved robot morphologies under simulation using a direct encoding and L-systems. The objects were ultimately fabricated by hand, and later, a three-dimensional printer. Additionally, Hornby and Pollack [13] used L-systems to evolve furniture designs, which were then manufactured by a three-dimensional printer. They found the generative encoding of L-systems produced designs faster and with higher fitness than a non-generative system. Generative systems are known to produce more compact encodings of solutions and thereby greater scalability than direct approaches (e.g., see [24]).

Lohn *et al.* [16] evolved and manufactured X-band satellite antenna for NASA’s ST5 spacecraft, representing the world’s first artificially evolved hardware in space. Significantly, the antenna’s performance was similar to a hand-designed antenna produced by an antenna-contractor. A generative encoding L-system was used which drew the three-dimensional model of the antenna when interpreted. The fitness was determined through the use of an antenna simulator.

Recently, the generative encoding Compositional Pattern Producing Networks (CPPNs) [26] have been used to evolve three-dimensional objects which were ultimately fabricated on a three-dimensional printer [1,2] [5]. Both interactive and target-based approaches were explored.

Significantly, most approaches have used simulations to provide the fitness scores of the evolved designs. Embodied evolutionary computing has typically referred to the existence of a physical solution in the fitness evaluation, and can be traced back to the origins of the discipline: the first Evolution Strategies were used to design jet nozzles as a string of real-valued diameters, which were then machined and tested for fitness [22]. Recently, Rieffel and Sayles [23] evolved circular two-dimensional shapes where each design was fabricated on a three-dimensional printer *before* assigning fitness values. Interactive evolution was undertaken wherein the fitness for each printed shape was scored subjectively. Each individual’s genotype consisted of twenty linear instructions which directed the printer to perform discrete movements and extrude the material. As a consequence of performing the fitness evaluation in the environment, i.e., after manufacture, the system as a whole can exhibit *epigenetic traits*, where phenotypic characteristics arise from the mechanics of assembly. One such example was found when selecting shapes that most closely resembled the letter ‘A’. In certain individuals, the cross of the pattern was produced from the print head dragging a thread of material as it moved between different print regions and was not explicitly instructed to do so by the genotype.

In this paper, we explore the evolution of a simple generative encoding to produce three-dimensional designs for manufacture on a three-dimensional printer.

## 2 Superformula

Gielis [9,10] found that the forms of a large variety of plants and other living organisms can be modelled by a single, simple, geometric equation, forming a generalisation of a hyper-ellipse, termed the *Superformula*. Modifying the set of real-valued parameters to the Superformula generates myriad and diverse natural polygons with corresponding degrees of freedom. The Superformula can be used to create three-dimensional objects, *Supershapes*, using the spherical product of two superformulas; in fact, by multiplying additional superformulas it can be extended to  $N$ -dimensions. “In general, one could think of the basic Superformula as a transformation to fold or unfold a system of orthogonal coordinate axes like a fan. This creates a basic symmetry and metrics in which distances can further be deformed by local or global transformations. Such additional transformations increase the plasticity of basic Supershapes” [10]. Gielis’ Superformula can be further generalised to increase the degrees of freedom, adding twist and further rotations, permitting the creation of more complex three-dimensional forms, including shells, möbius strips, and umbilic tori. Gielis’ Superformula, which defines a supershape in 2 dimensions is given in the following equation, where  $r$  is the radius;  $\phi$  is the angle;  $a > 0$ ,  $b > 0$  control the size of the supershape and typically = 1; and  $m$  (symmetry number),  $n_1$ ,  $n_2$  and  $n_3$  (shape coefficients) are the real-valued parameters:

$$r = f(\phi) \frac{1}{\sqrt[n_1]{(|\frac{1}{a}\cos(\frac{m}{4}\phi)|)^{n_2} + (|\frac{1}{b}\sin(\frac{m}{4}\phi)|)^{n_3}}} \quad (1)$$

Using the spherical product, the extension to three dimensions:

$$x = r_1(\theta) \times \cos(\theta) \times r_2(\varphi) \times \cos(\varphi)$$

$$y = r_1(\theta) \times \sin(\theta) \times r_2(\varphi) \times \cos(\varphi)$$

$$z = r_2(\varphi) \times \sin(\varphi)$$

Where  $-\pi/2 \leq \varphi \leq \pi/2$  for latitude and  $-\pi \leq \theta \leq \pi$  for longitude.

Example shapes generated with the Superformula can be seen in Figure 1 where the cube, star, and heart can be generated from the same set of eight real-valued parameters; the torus requiring two additional parameters; the shell a total of twelve; and the möbius strip a total of fifteen.

A “Superdupershape explorer” and its source code, licensed under Creative Commons Attribution-Share Alike 3.0 and GNU GPL license, can be found at <http://openprocessing.org/visuals/?visualID=2638>.

Given a target shape it is often very useful to identify a representative formula. Optimisation methods, such as the Levenberg-Marquardt (LM) theory [20], have typically been used to identify the best fitting supershape parameters (e.g., [11]). However LM cannot retrieve all of the parameters required for supershape fitting. Bokhabrine *et al.* [4] used a Genetic Algorithm (GA) [12] to evolve all supershape parameters for surface reconstruction (i.e., a target-based

approach) using an inside-outside function [7] for fitness computation. Voisin *et al.* [27] later extended this to utilise a pseudo-Euclidean distance for fitness determination, yielding improved performance. Additionally, Morales *et al.* [18] used a GA to evolve  $N$ -dimensional Superformula for clustering.

### 3 Target-Based Evolution of Supershapes

The cube, star, and heart shapes (as seen in Figure 1) are here converted into 125,000 voxel binary three-dimensional grids and used as the desired targets, where the fitness of an individual is the fraction of voxels that correctly match. The genotype of each individual in the population consists of eight real-valued parameters in the range  $[0,20]$  which affect the Superformula, giving rise to the Supershape. The GA proceeds with a population of 200 individuals, a per allele mutation rate of 25%, and mutation step size of  $\pm rand(5)$ , where *rand* selects a real-valued number in the range  $[0,5]$ ; the GA tournament size for both selection and replacement is set to 3.

Figure 2 shows the fraction of total voxels matched to the target shapes during evolution of the Supershapes; results presented are an average of 10 experiments. Similar to [5], a large number of voxels are quickly matched, however here the target object is not identifiable until approximately 99% are set correctly. As such, the small differences in fitness between the treatments represent substantial differences in whether the target object is recognisable. In all cases, greater than 99.5% fitness is achieved. From Figure 2a it can be seen that, on average, the GA takes approximately 1100 evaluations to reach >99% matching voxels of a target cube object and 3700 evaluations to achieve >99.9%. Figure 2b shows that on average approximately 3900 evaluations are required to reach >99% matching voxels of a target star object and 16100 evaluations to achieve >99.5%. Finally, Figure 2c shows that, on average, >99% matching voxels of a target heart object is reached after 6400 evaluations and >99.5% after 24000 evaluations.

At the end of the experiments, the fittest individual was subsequently fabricated by a three-dimensional printer and can be seen in Figure 3, including the supporting rafts required for manufacture. Figure 4 illustrates a sample of the evolved individuals from one cube experiment, Figure 5 similarly for the star experiment, and Figure 6 for the heart experiment.

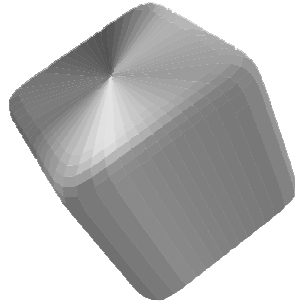
### 4 Conclusions

This paper has shown that it is possible to evolve a vector of reals which are used as Superformula parameters to generate three-dimensional objects. Target-based evolution was used to explore the ability of Superformula to create complex objects, particularly those that resemble natural designs. The experiments showed that with target-based evolution very closely matching objects can be identified. One significant advantage of the approach over alternative representations is the simplicity and compactness of the encoding.

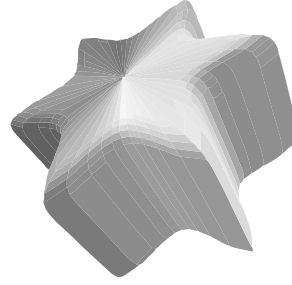
## References

1. Auerbach, J.E., Bongard, J.C.: Dynamic resolution in the co-evolution of morphology and control. In: 12th International Conference on the Synthesis and Simulation of Living Systems. pp. 451–458. ALife XII, MIT Press (2010)
2. Auerbach, J.E., Bongard, J.C.: Evolving CPPNs to grow three-dimensional physical structures. In: Proceedings of the 12th annual conference on Genetic and Evolutionary Computation. pp. 627–634. GECCO '10, ACM (2010)
3. Bentley, P.: Generic Evolutionary Design of Solid Objects using a Genetic Algorithm. Ph.D. thesis, University of Huddersfield (1996)
4. Bokhabrane, Y., Fougerolle, Y.D., Foufou, S., Truchetet, F.: Genetic algorithms for Gielis surface recovery from 3D data sets. In: Image Processing, 2007. ICIP 2007. IEEE International Conference on. vol. 2, pp. 549–552. IEEE Press (2007)
5. Clune, J., Lipson, H.: Evolving three-dimensional objects with a generative encoding inspired by developmental biology. In: Proceedings of the European Conference on Artificial Life. pp. 141–148. ECAL '11, MIT Press (2011)
6. Eggenberger, P.: Evolving morphologies of simulated 3D organisms based on differential gene expression. In: Proceedings of the Fourth European Conference on Artificial Life. pp. 205–213. MIT Press (1997)
7. Fougerolle, Y.D., Gribok, A., Foufou, S., Truchetet, F., Abidi, M.A.: Supershape recovery from 3D data sets. In: Image Processing, 2006. ICIP 2006. IEEE International Conference on. pp. 2193–2196. IEEE Press (2006)
8. Funes, P., Pollack, J.: Evolutionary body building: Adaptive physical designs for robots. *Artificial Life* 4, 337–357 (October 1998)
9. Gielis, J.: A generic geometric transformation that unifies a wide range of natural and abstract shapes. *American Journal of Botany* 90, 333–338 (2003)
10. Gielis, J., Beirinckx, B., Bastiaens, E.: Superquadrics with rational and irrational symmetry. In: Proceedings of the eighth ACM symposium on Solid modeling and applications. pp. 262–265. SM '03, ACM (2003)
11. Gupta, A., Bajcsy, R.: Volumetric segmentation of range images of 3D objects using superquadrics models. *Computer Vision, Graphics and Image Processing: Image Understanding, CVGIP* 58, 302–326 (1993)
12. Holland, J.H.: *Adaptation in Natural and Artificial Systems*. University of Michigan Press (1975)
13. Hornby, G.S., Pollack, J.B.: The advantages of generative grammatical encodings for physical design. In: *Evolutionary Computation, 2001. Proceedings of the IEEE Congress on*. pp. 600–607. CEC 2001, IEEE Computer Society Press (2001)
14. Jacob, C.J., Hushlak, G.: Evolutionary and swarm design in science, art, and music. In: *The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music*, pp. 145–166. Springer (2007)
15. Jacob, C.J., Nazir, A.: Polyhedron evolver—evolution of 3D shapes with evolva. In: *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics: Volume VII Information Systems Development II*, July 14–18. International Institute of Informatics and Systemics (2002)
16. Lohn, J.D., Hornby, G.S., Linden, D.S.: Human-competitive evolved antennas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 22, 235–247 (August 2008)
17. McGuire, F.: The origins of sculpture: Evolutionary 3D design. *IEEE Computer Graphics Applications* 13, 9–11 (January 1993)

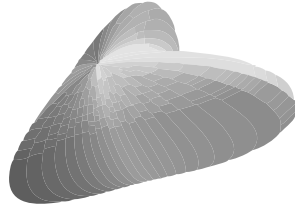
18. Morales, A.K., Bobadilla, E.A.: Clustering with an N-dimensional extension of Gielis Superformula. In: Proceedings of the 7th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Databases. pp. 343–350. World Scientific and Engineering Academy and Society (WSEAS) (2008)
19. Pollack, J.B., Lipson, H., Hornby, G., Funes, P.: Three generations of automatically designed robots. *Artificial Life* 7, 215–223 (2001)
20. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes in C: The Art of Scientific Computing* (Second Edition). Cambridge University Press (1992)
21. Prusinkiewicz, P.W., Lindenmayer, A.: *The Algorithmic Beauty of Plants*. Springer (1990)
22. Rechenberg, I.: *Evolutionstrategie – Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Ph.D. thesis, Technical University of Berlin (1971)
23. Rieffel, J., Sayles, D.: EvoFAB: A fully embodied evolutionary fabricator. In: Tempesti, G., Tyrrell, A.M., Miller, J.F. (eds.) *Evolvable Systems: From Biology to Hardware - 9th International Conference*. pp. 372–380. ICES 2010, Springer (2010)
24. Schoenauer, M.: Shape representations and evolution schemes. In: *Proceedings of the 5th Annual Conference on Evolutionary Programming*. pp. 121–129. MIT Press (1996)
25. Sims, K.: Evolving 3D morphology and behavior by competition. *Artificial Life* 1, 353–372 (1994)
26. Stanley, K.: Compositional pattern producing networks: A novel abstraction of development. *Genetic Programming and Evolvable Machines* 8, 131–162 (2007)
27. Voisin, S., Abidi, M.A., Foufou, S., Truchetet, F.: Genetic algorithms for 3D reconstruction with supershapes. In: *Image Processing, 2009. ICIP 2009. IEEE International Conference on*. pp. 529–532. IEEE Press (2009)
28. Watabe, H., Okino, N.: A study on genetic shape design. In: *Proceedings of the 5th International Conference on Genetic Algorithms*. pp. 445–451. ICGA '93, Morgan Kaufmann Publishers Inc. (1993)



(a) Cube  $m_1 = 4$ ,  $n_{1,1} = 10$ ,  $n_{1,2} = 10$ ,  $n_{1,3} = 10$ ,  $m_2 = 4$ ,  $n_{2,1} = 10$ ,  $n_{2,2} = 10$ ,  $n_{2,3} = 10$



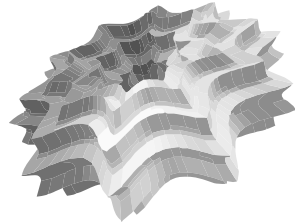
(b) Star  $m_1 = 6$ ,  $n_{1,1} = 5$ ,  $n_{1,2} = 10$ ,  $n_{1,3} = 10$ ,  $m_2 = 4$ ,  $n_{2,1} = 10$ ,  $n_{2,2} = 10$ ,  $n_{2,3} = 10$



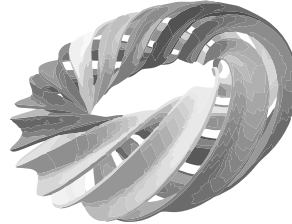
(c) Heart  $m_1 = 3$ ,  $n_{1,1} = 1.5$ ,  $n_{1,2} = 12$ ,  $n_{1,3} = 3$ ,  $m_2 = 0$ ,  $n_{2,1} = 3$ ,  $n_{2,2} = 0$ ,  $n_{2,3} = 0$



(d) Shell  $m_1 = 3$ ,  $n_{1,1} = 1.5$ ,  $n_{1,2} = 12$ ,  $n_{1,3} = 3$ ,  $m_2 = 0$ ,  $n_{2,1} = 3$ ,  $n_{2,2} = 0$ ,  $n_{2,3} = 0$ ,  $t_2 = 2$ ,  $d_1 = 1$ ,  $d_2 = 1$ ,  $c_1 = 5$



(e) Torus  $m_1 = 10$ ,  $n_{1,1} = 10$ ,  $n_{1,2} = 10$ ,  $n_{1,3} = 10$ ,  $m_2 = 10$ ,  $n_{2,1} = 10$ ,  $n_{2,2} = 10$ ,  $n_{2,3} = 10$ ,  $t_1 = 2$ ,  $c_3 = 0$



(f) Möbius Strip  $m_1 = 3$ ,  $n_{1,1} = 1.5$ ,  $n_{1,2} = 12$ ,  $n_{1,3} = 3$ ,  $m_2 = 0$ ,  $n_{2,1} = 3$ ,  $n_{2,2} = 0$ ,  $n_{2,3} = 0$ ,  $t_1 = 4$ ,  $t_2 = 0$ ,  $d_1 = 0$ ,  $d_2 = 0$ ,  $c_1 = 5$ ,  $c_2 = 0.3$ ,  $c_3 = 2.2$

Fig. 1: Example three-dimensional Supershapes

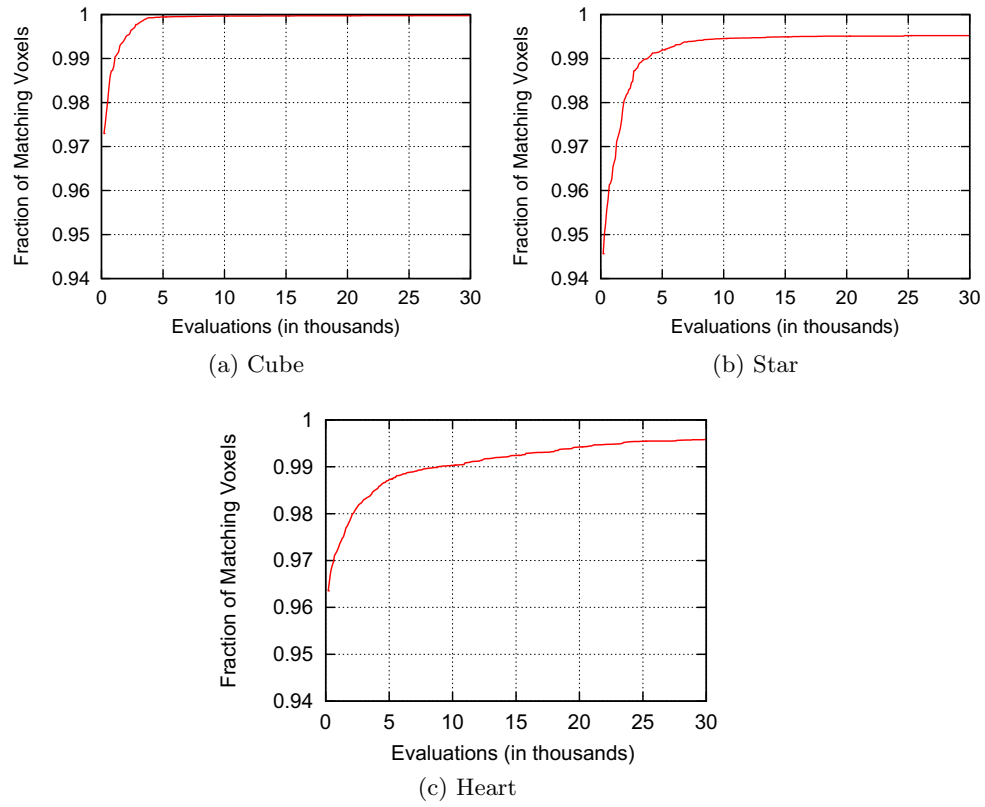


Fig. 2: Evolution of three-dimensional Supershapes

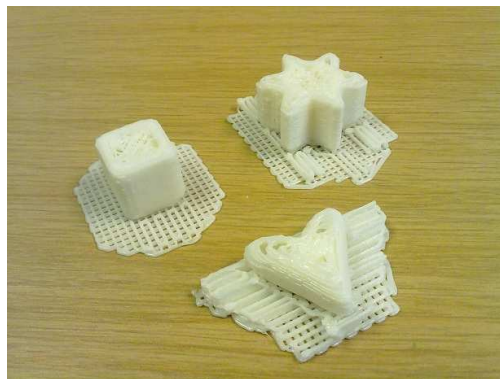


Fig. 3: Cube, Star and Heart fabricated by a three-dimensional printer.



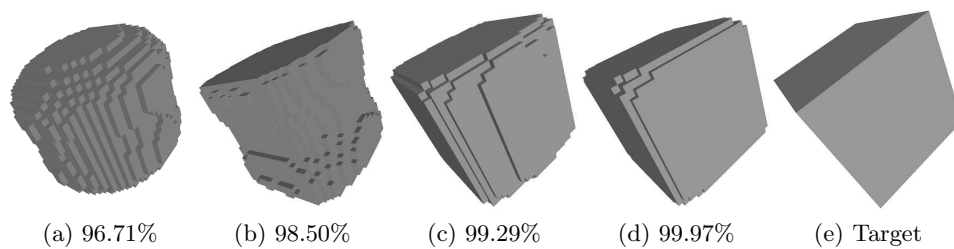


Fig. 4: Evolution of a three-dimensional cube

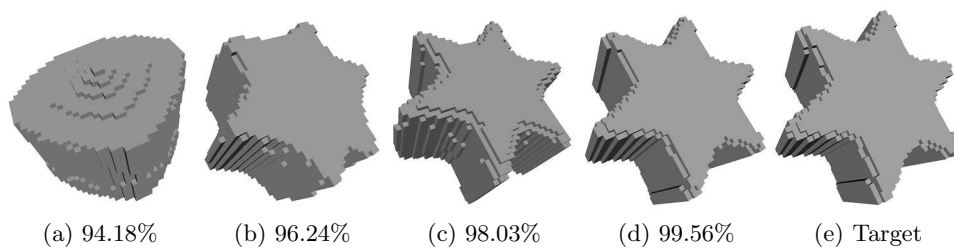


Fig. 5: Evolution of a three-dimensional star

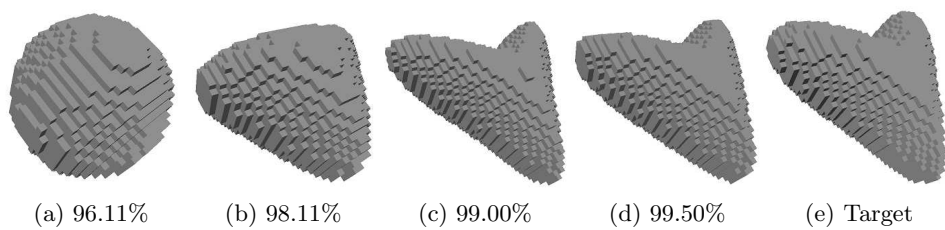


Fig. 6: Evolution of a three-dimensional heart